



University of Toronto

APS105H1F — Computer Fundamentals

Lab 3: Functions

Introduction

This lab is designed to give you practice using and writing functions in C++. You will also gain more experience using other C++ statements that you have learned in the course.

This lab has three parts. By the end of the lab session, you will have created three files, namely files `Lab3Part1.c`, `Lab3Part2.c`, and `Lab3Part3.c`. Please keep in mind that the marking process will begin approximately forty minutes before the end of the lab.

Before attempting this lab, you should review your lecture notes from APS105 up to and including the lecture on Friday September 29th and also ensure that you have read Chapter 1, Chapter 2, and Sections 4.2 and 4.3 from the Savitch text.

You should try to complete the lab in advance of your practical session and get help from a TA soon after arriving at SF 1013 so that you may promptly overcome any remaining difficulties.

Part 1 — Simple Calculations

In a file named `Lab3Part1.c`, write a program that asks the user to enter a number, reads the entered value into a variable named `x` of type `double`, and then performs each of the calculations described below that involve the value in `x`. Your program should produce informative output for each of the calculations, with one result per line. For **each** calculation, if the mathematical result is not defined, your program should print the result `Not defined` and a short explanation as to why the result is not defined. Assume that all of the mathematical results should be real-valued, so for example, attempting to compute $\sqrt{-42}$ should result in a message like:

```
Square root of -42 is: Not defined because -42 is < 0.
```

The prompt to the user should be: `Enter a value for x:`

Each of the following calculations are to be performed:

1. \sqrt{x}
2. $\log_{10}(\sqrt{x^2 + 1} - x)$
3. x^{14}
4. $e^{14 \log_e(x)}$
5. $\sin^2(x) + \cos^2(x)$
6. $1/x^2$

Part 2 - Computing GCDs

The Greatest Common Divisor (GCD) of two nonnegative integers x and y is the largest integer that divides evenly into both x and y . For example, the GCD of 49 and 35 is 7, the GCD of 6 and 18 is 6, the GCD of 32 and 33 is 1, and the GCD of 0 and 42 is 42. We shall say that the GCD of 0 and 0, the GCD of -7 and 3 and the GCD of -6 and -2 are not defined. The GCD function is symmetric in that the GCD of a and b is the same as the GCD of b and a .

One way to compute the GCD is as follows: Let m and n be variables containing two **positive** integers whose GCD is to be computed. Divide m by n . Save the divisor in m and save the remainder in n . If n is 0, stop the calculation as the variable m will contain the GCD. Otherwise, repeat the process, starting with the division of m by n . (We say that integer a divides evenly into b if the remainder from the division is 0. In this case, there exists another integer c such that $b = c \times a$.)

Your task is to write a C++ program that prompts its user to enter two integer values and prints out their GCD or a message indicating that the GCD is not defined.

Before writing your program, ensure that you understand the algorithm for computing GCDs by tracing it on a few examples. For example, trace it to compute the GCD of 49 and 35, the GCD of 6 and 18 and the GCD of 33 and 1.

Part 3 - A GCD function

Using your solution to Part 2, write a C++ function that takes two integer parameters and returns their GCD or the value -1 in the case that the GCD is not defined. If you have not completed Part 2, write your function so that it always returns the result 42 when the GCD is defined or the value -1 when the GCD is not defined.

Functions of this nature should not print output on the user's display. They should instead communicate with the invoking function through the return value.

Write a short `main` function that demonstrates the use of your GCD function. Your `main` function should prompt the user for two integers, read the values, call your GCD function and print the numerical result returned.